

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

**System and Method for Synchronizing Streaming
Content with Enhancing Content Using Pre-announced
Triggers**

Inventor(s):

David Feinleib
Isaac A. Sheldon

ATTORNEY'S DOCKET NO. MS1-288US

09423205

1 **TECHNICAL FIELD**

2 This invention relates to a system and method for synchronizing streaming
3 content with enhancing content. More particularly, this invention relates to a
4 system that announces transmission of upcoming enhancing content that is to be
5 associated with the streaming content, such as video or a television signal. The
6 system further encompasses timely reception of the enhancing content and
7 displaying it in conjunction with the streaming content through use of triggers.

8
9 **BACKGROUND**

10 Video content programs, such as TV shows and pay-per-view movies, are
11 delivered to a viewer as a continuous data stream. Programs are most commonly
12 distributed using a wireless broadcast system or a cable system. In the first
13 instance, the programs are broadcast over a wireless network and received at
14 individual homes through an antenna or satellite dish. In the latter case, the
15 programs are transmitted over cable to set-top boxes resident in the viewers'
16 homes.

17 In traditional broadcast distribution systems, there is no opportunity for
18 interactive control of the content by the viewer. The viewer simply has an option
19 to watch the program, change to a different program, or turn off the television.
20 However, as consumers have learned from playing video games on their
21 televisions, non-interactive viewing is not nearly as fun or sensory rich as
22 interactive entertainment.

23 To enhance the traditional way of viewing television, there has been some
24 effort toward the production of interactive programming content. As presently
25 contemplated, additional interactive content is created to enhance the existing

1 traditional program. This supplemental content is played along with the
2 continuous video stream to enable viewers to interact with the program in a more
3 involved manner than simply watching it. The supplemental content might, for
4 example, ask the viewer questions about the episode, or play games with the
5 viewer that relate to the show, or describe behind-the-scenes aspects of making the
6 program, or provide links to stores that sell merchandise sponsored by the show.
7 In addition, the content may not be tied to a particular program, but instead be used
8 to convey general information, such as tickers for news headlines, weather
9 information, sports scores, and so forth.

10 The Internet is rapidly emerging as a means for supplying interactive
11 content. Resources available on the Internet are most commonly presented as
12 HTML (hypertext markup language) pages. Users can browser through pages of
13 information via the World Wide Web ("Web") and receive multimedia-like
14 experiences with video, audio, images, text, and other multimedia rich resources.
15 Newer technologies are combining the television and Internet experiences to
16 provide interactive content. Set-top boxes, for example, are being configured to
17 support Web browsers that enable access to Web content in addition to receiving
18 traditional television signals. As an example, WebTV Networks, Inc.
19 manufactures and sells a browser-based set-top box (STB) having a television
20 tuner, a browser, and a standard network interface device that enables
21 simultaneously viewing of both television programs and web programming.

22 Newer generations of personal computers (PCs) are being implemented
23 with broadcast reception capabilities, allowing them to receive and display
24 television signals. Such PCs are equipped with a television tuner and software for
25 television reception. A Windows-brand operating system from Microsoft

1 Corporation, for example, includes software that supports a number of interactive
2 services and reception of broadcast data and content feeds.

3 These newer interactive television technologies combine the power of
4 traditional television with the compelling interactivity of the Internet. One
5 common approach today is to provide the enhancing content via the vertical
6 blanking interval (VBI) of the streaming television content. The VBI is a non-
7 viewable portion of the television signal. In the future, enhancing content may
8 also be provided over satellites (e.g., DVB, DSS, etc.) and as part of digital
9 television (DTV) feeds.

10 Unfortunately, there remains a significant hurdle concerning creation and
11 distribution of interactive programs. TV programs are typically broadcast as a
12 continuous data stream. Attempting to synchronize the enhancing content,
13 regardless of how that content is delivered (i.e., via the VBI, satellite feeds, etc.),
14 with the streaming content poses a difficult design challenge.

15 The inventors have developed a system and method that address this
16 challenge.

17 **SUMMARY**

18 This invention concerns a client-server architecture that synchronizes
19 streaming content with enhancing content via pre-announced triggers.

20 According to one implementation, the architecture includes server-side
21 components that form announcements containing information specifying how and
22 when to receive upcoming triggers. The announcements contain such information
23 as an identification of the sender, a broadcast locator (e.g., URL, channel,
24 frequency, etc.) at which the triggers and enhancing data files are to be broadcast
25

1 or multicast, a time when the triggers and data files are to be sent, and a protocol
2 used to transmit the triggers and data files. The server transmits the
3 announcements to a general broadcast or multicast IP (Internet Protocol) address.

4 Client-side components include a listener to monitor the IP address for the
5 announcements and a filter to extract selected ones of the announcements.
6 Selected announcements are stored in an electronic programming guide (EPG)
7 database in correlation with the streaming content programs to indicate that the
8 associated programs are interactive.

9 When a user tunes to an interactive program, the client opens a container
10 HTML page that contains controls to receive the streaming content program and to
11 extract the announcements from the guide database. One of the controls monitors
12 an IP address at particular times specified by a selected announcement to receive
13 the triggers corresponding to the interactive program.

14 The server delivers the triggers at times synchronized to the streaming
15 content. When triggers arrive, a client control processes the triggers to coordinate
16 presentation of the enhancing content with the streaming content. This processing
17 may involve invoking one or more scripts in the container page or performing
18 general navigation in the container page.

19 The client-server architecture also enables delivery of headlines and other
20 data items employed in a ticker using pre-announced and timely broadcast triggers.
21 The data items are carried by the triggers and accumulated into an array. The
22 container page inserts the array into a displayed ticker.

1 **BRIEF DESCRIPTION OF THE DRAWINGS**

2 Fig. 1 is a diagrammatic illustration of a client-server architecture for
3 synchronizing streaming content with enhancing content using pre-announced
4 triggers.

5 Fig. 2 is a block diagram of an exemplary client computing unit.

6 Fig. 3 shows an exemplary configuration of a client software architecture
7 implemented on the client computing unit.

8 Fig. 4 shows a flow diagram showing steps in a method for synchronizing
9 streaming content with enhancing content through use of announcements and
10 triggers.

11 Fig. 5 shows an exemplary HTML page that combines streaming content
12 with enhancing content. It also illustrates use of triggers to carry content for
13 tickers.

14
15 **DETAILED DESCRIPTION**

16 This invention concerns an architecture for synchronizing streaming content
17 with enhancing content. The architecture governs announcement of upcoming
18 enhancing content, reception of the enhancing content when it arrives, and
19 association of the enhancing content with the streaming content using triggers

20 The enhancing content consists of a set of data files and triggers that are
21 multicast to a pre-defined network address to cause specific actions or events to
22 occur on any listening client. The triggers are tied to the associated streaming
23 content. The enhancing content can take many forms, including text, images,
24 animated images, audio, hyperlinks, controls, executable code, and so forth.

1 Announcements are sent ahead of the triggers to announce when the triggers will
2 be broadcast and on what multicast address.

3 Another aspect of this invention concerns a way to receive headlines and
4 other data items as triggers and accumulate them into a ticker.

5 6 General Architecture

7 Fig. 1 shows an architecture 20 that enables delivery of streaming content
8 and enhancing content from one or more content providers, as represented by a
9 primary content provider 22(1) and a secondary content provider 22(2), to multiple
10 clients 24(1), 24(2), 24(3), and 24(4). The Fig. 1 illustration shows many different
11 ways to deliver streaming and enhancing content. One way is for the same
12 provider to deliver both forms of content directly to the clients over the same
13 distribution network. This situation is represented by content provider 22(1)
14 serving streaming and enhancing content over network 26 to clients 24(1) and
15 24(2).

16 Another way is for different providers to deliver the contents separately
17 over the same or different networks. This situation is represented by the primary
18 content provider 22(1) serving streaming content over network 26 to clients 24(1)
19 and 24(2) and the secondary content provider 22(2) serving enhancing content
20 over network 28 to the clients 24(1) and 24(2).

21 Still another way is for a content provider to deliver its content to an
22 intermediary broadcast center, which broadcasts the content to clients. This third
23 approach is represented by either content provider 22(1) or 22(2) delivering
24 packets to a broadcast center 30, which converts network data packets to broadcast
25

1 packets and broadcasts the packets over a broadcast medium 32 to clients 24(3)
2 and 24(4).

3 The networks 26, 28, and 32 represent many diverse types of networks
4 ranging from data networks (e.g., Internet, LAN (local area network), WAN (wide
5 area network), etc.) to television networks (e.g., cable, satellite, RF, microwave,
6 etc.). Technologies used to implement the networks are well known and include
7 such technologies as wire-based technologies (e.g., fiber optic, cable, wire, etc.),
8 wireless technologies (e.g., satellite, RF, etc.), and switching technologies (e.g.,
9 ATM (Asynchronous Transfer Mode), Ethernet, etc.). These technologies employ
10 different data communication protocols (e.g., TCP/IP, IPX/SPX, DSS, NABTS,
11 etc.). One specific implementation involves IP over any one of these protocols
12 (e.g., IP/NABTS, IP/DSS, IP/DVB, etc.). For an example of IP/NABTS, please
13 refer to co-pending US Patent Application Serial Number 08/726,529, entitled
14 "Method for Sending Computer Network Data As Part Of Vertical Blanking
15 Interval", which was filed October 7, 1996.

16 The primary content provider 22(1) has a streaming content server 34 to
17 serve streaming content (e.g., video and audio) to the clients. The content may be
18 in many different forms, such as TV programs, movies, animation, music, news,
19 documentaries, educational productions, and the like. As an exemplary
20 implementation, the streaming content server 34 may be implemented as personal
21 computers or workstations running a multitasking, disk-based operating system,
22 such as Windows NT from Microsoft Corporation, that are configured as
23 continuous media file servers that serve data files at a constant data rate. As an
24 alternative to a content server, the primary content provider may be a broadcast
25 headend that broadcasts the primary content (e.g., broadcast or satellite TV).

The primary content provider 22(1) has an enhancing content server 36 to serve supplemental or enhancing content to the clients in order to enhance the streaming content served by the streaming content server 34. The enhancing content could alternatively be stored with the primary content and served before, along with, or after the primary content. For instance, a single server may serve both the primary and enhancing content and record the content onto a storage medium for later transmission.

The enhancing content may be text, graphics, video, pictures, sound, or other multimedia types, as well as applications or other executable code. Examples of enhancing content include trivia questions or games related to the program, advertisements, merchandise or other memorabilia, hyperlinks to similar programs of similar type or starring the same actor/actress, and so on.

In the implementation described herein, the enhancing content is constructed as a hypertext file, or more particularly as an HTML document (or "Web page") which can be rendered by a browser. The HTML document may include links to other target resources that supply even more content. In concept, the target resource can be virtually any type of object—including executable programs, text or multimedia documents, sound clips, audio segments, still images, computers, directories, and other hyperlinks. In most Web pages, hyperlink targets are files that reside on computers connected to the Internet. However, a hyperlink target can also be a particular location within a document, including the document that is currently being rendered, as well as to other files that may be locally stored at the client.

The primary content provider 22(1) also implements an announcement generator 38 that generates and transmits announcements to notify clients of

1 upcoming transmission of enhancing content. Through the announcements, the
2 providers tell the clients what data will be served over the network at a given time
3 and how to find that data.

4 The secondary content provider 22(2) is similar to the primary content
5 provider 22(1) in that it implements an enhancing content server 40 and an
6 announcement generator 42. These components function essentially as described
7 above with respect to the primary content provider 22(1).

8 9 **Enhancing Content Stream**

10 The content providers serve both streaming content and enhancing content
11 to the clients. The architecture employs three elements to implement reception and
12 display of enhancing content: announcements, triggers, and data files. The first
13 element is an "announcement" that describes the IP address and port on which
14 corresponding triggers will be sent for a particular enhancement. The second
15 element is a "trigger", which tells the client to perform particular actions (e.g.,
16 execute a script). The third element is the "data files", which are either packages
17 containing interactive content files or the interactive content files themselves.
18 Together, these elements form the enhancement stream that is delivered to the
19 client.

20 Announcements, triggers, and data files are all sent on separate IP
21 addresses/ports. Sending an enhancement stream involves transmitting the
22 following elements:

- 23
- 24 1. An announcement indicating a trigger address and port.
- 25 2. A special trigger indicating an address and port for file transmission.

- 1 3. A data file containing interactive content files or individual files.
- 2 4. If necessary, a trigger causing the interactive content to be extracted
- 3 out of a container file.
- 4 5. A trigger to cause the interactive content to be displayed.

5

6 “Announcements” inform the client that the streaming content, such as a

7 television show, is interactive. The announcements provide details about the

8 enhancements, the show that the content enhances, and the page that contains the

9 enhancement's starting point. More particularly, the announcements include

10 transmission-related information, such as an identification of the sender, a locator

11 (e.g., URL, channel, frequency, etc.) at which the triggers and data files are to be

12 transmitted, a time when the triggers and data files are to be sent, and a protocol

13 used to transmit the triggers and data files. The announcements might further

14 include information pertaining to the content of the transmission, such as a title, a

15 type of content (e.g., sports, science fiction, mystery, action, documentary, audio,

16 graphical, etc.), a subject matter description, a length of transmission, a rating,

17 actor/actress names, and so forth.

18 Announcements are based on the Internet standard Session Description

19 Protocol (SDP). An announcement is made up simply of lines of text, with

20 identifiers on the left of an equal sign and associated text on the right. An

21 announcement can be stored as a text file until it is ready to be broadcast.

22 There are several types of enhancement announcements. The most

23 common types are current announcements, which refer to a show that is currently

24 on, and channel announcements, which refer to a given channel. Table 1 illustrates

25 a sample announcement.

Table 1

<u>Example</u>	<u>Description</u>
v=0	SDP Version number, must be 0.
o=enhID 2 1 IN IP4 155.55.55.55	“enhID” specifies a user identifier. “2” is the session identifier. “1” is the version identifier. “155.55.55.55” is replaced with the IP address of the server transmitting or encoding the content. As a whole this field serves as an identifier for an enhancement.
s=Title	Contains the title for the enhancement.
c=IN IP4 233.17.43.44/3	Indicates the IP address on which triggers will be sent.
t=2208988801 2208990601	Start and finish time for an enhancement.
a=EnhID:{9E2E8B20-083E-11d1-898F-00C04FBBDEBC}	Identifies the announcement as an enhancement announcement (as opposed to some other type of announcement).
m=data 17832 udp 0	Specifies the port used for sending triggers.

“Triggers” are notifications that are sent to clients at specific times during an interactive data stream. Triggers cause an action to occur on a client, such as causing a script to be executed. In this manner, the enhancing content invoked by the triggers is synchronized with the streaming content.

Each type of trigger has a different format. However, all triggers are formatted as strings containing three parts: a key or numerical identifier, the

1 trigger data, and a checksum value. The key is separated from the data by white
2 space, typically a space or tab character. The format of a trigger is as follows:

3
4 Key TriggerData[checksum]

5
6 where "Key" is the numerical identifier, "TriggerData" is the trigger data, and
7 "checksum" is the checksum value. For example, the following string might be
8 transmitted to the client for a navigation or "Nav" trigger:

9 "2 bpc://MSNBC/default.htm[BA7E] "

10
11 The preceding example illustrates the syntax of the Nav trigger as it is
12 transmitted to a client. This is not the format typically used in writing stream
13 language files. The syntax transmitted to clients contains additional information
14 that you do not need to specify, namely a checksum value and a fully specified
15 Broadcast Architecture URL. The Broadcast Architecture supports several basic
16 types of triggers as shown in Table 2.
17
18
19
20
21
22
23
24
25

Table 2

<u>Key</u>	<u>Name</u>	<u>Description</u>
0	Error	No action is taken.
1	Data	Receives data using the specified transfer protocol.
2	Nav	Displays a page. If a target is specified, for example by using a FRAME or DIV tag set, the page is displayed in that target.
3	Nav	Displays a page. This key value has the same functionality as the previous trigger and has been kept for compatibility with earlier versions.
4	Script	Runs the specified script.
6 – 999	Reserved	Reserved for future use.
1,000 and up	UserTrigger	Sends a user-defined trigger event.

Table 3 lists exemplary triggers for the navigation-type, script-type, and data-type triggers specified by the Broadcast Architecture.

Table 3

<u>Trigger Type</u>	<u>Description</u>	<u>Format</u>
<i>Navigation</i>		
Top Page Navigation	"Home page" for enhancements	"2 <bpc://MyEnhancement/toppage.htm>"
Target Navigation	To navigate to a named target	"2 <itv.htm>[targ:LOGO_FRAME]"
<i>Script</i>		
	Execute ECMAScript (Jscript) in the top page	"4 alert('Hi!');" "4 Navigate('VIEW_FRAME', 'http://www.microsoft.com');"
<i>Data</i>		
FTS	Starts listening for FTS data	"1 <FTS> 233.17.43.44:2000&MyEnhancement"
CAB	Unpacks cabinet (CAB) file	"1 <CAB> bpc://MyEnhancement/toppage.cab"

Enhancement data files and their dependencies are transmitted to clients using a transfer service. One exemplary service is the File Transfer Service (FTS), which is a component of NetShow, a Microsoft product used for streaming video and data on the Internet and on Intranets using IP Multicast. FTS sends files using a transfer mechanism based on IP multicast that includes forward error correction (FEC). However, other file transfer services might be employed as well. The architecture allows use of different file transfer services as needed.

Enhancement pages, like Web pages, are typically composed of several files. For example, the enhancement file "MyEnh.htm" might contain graphic images in Graphics Interchange Format (GIF) and Joint Photographic Experts Group format (JPEG), ActiveX controls, FutureSplash animations, and other such

1 files that must be downloaded for the enhancement page to display properly.
2 These additional files are referred to as "dependency file" or "dependencies" of the
3 enhancement file.

4 The enhancement dependency files are stored on the client before the
5 enhancement is displayed. Otherwise, the user may end up viewing an incomplete
6 enhancement and will perceive the content as broken. There are several ways of
7 handling dependencies. One approach is to allocated broadcast bandwidth and
8 repeatedly download the dependency files. Another approach is to require the user
9 to explicitly download the dependency files, either from a Web site or a portable
10 medium.

11 Still another approach is to package the enhancement and all its
12 dependencies together in a single file. In one implementation, this involves use of
13 cabinet (CAB) files. Packing the enhancement files together with its dependencies
14 ensures that the user receives a complete all-or-nothing viewing experience
15 because they either get it all or nothing, which may be better than getting bits and
16 pieces.

17 18 **Client System**

19 Clients 24(1)-24(4) can be implemented in a number of ways, including
20 computers that are broadcast enabled, set-top boxes, and computer enhanced
21 television units.

22 Fig. 2 shows an exemplary configuration of a client 24(1) implemented as a
23 broadcast-enabled computer. It includes a central processing unit 50 having a
24 processor 52, volatile memory 54 (e.g., RAM), and program memory 56 (e.g.,
25 ROM, disk drive, floppy disk drive, CD-ROM, etc.). The client 24 (1) has one or

1 more input devices 58 (e.g., keyboard, mouse, etc.), a computer display 60 (e.g.,
2 VGA, SVGA) to display the video content and enhancing content, and a stereo I/O
3 62 for interfacing with a stereo system to play audio content.

4 The client 24(1) includes a digital broadcast receiver 64 (e.g., satellite dish
5 receiver, RF receiver, microwave receiver, network connection, etc.) and a tuner
6 66 which tunes to frequencies of a broadcast network. The tuner 66 is configured
7 to receive digital broadcast data in a particularized format, such as MPEG-encoded
8 digital video and audio data, as well as digital data in many different forms,
9 including software programs and programming information in the form of data
10 files. The client 24(1) also has a modem 68 that provides access to the Internet or
11 other network. For other implementations, the modem 68 might be replaced by a
12 network card, an RF receiver, or other type of port/receiver.

13 One example implementation of a broadcast-enabled PC is described in a
14 co-pending U.S. Patent Application Serial No. 08/503,055, entitled "Broadcast-
15 Enabled Personal Computer," filed January 29, 1996 in the names of Gabe L.
16 Newell, Dan Newell, Steven J. Fluegel, David S. Byrne, Whitney McCleary,
17 James O. Robarts, Brian K. Moran; William B. McCormick, T.K. Backman,
18 Kenneth J. Birdwell, Joseph S. Robinson, Alonzo Garipey, Marc W. Whitman,
19 and Larry Brader. This application is assigned to Microsoft Corporation, and is
20 incorporated herein by reference.

21 The client 24(1) runs an operating system 70 that supports multiple
22 applications. The operating system is preferably a multitasking operating system
23 that allows simultaneous execution of multiple applications. One preferred
24 operating system is a Windows brand operating system sold by Microsoft
25 Corporation, such as Windows 98, Windows CE, or Windows NT or other

1 derivative versions of Windows. It is noted, however, that other operating systems
2 may be employed, such as the Macintosh operating system from Apple Computer,
3 Inc. and the OS/2 operating system from IBM.

4 The client 24(1) further runs a Web browser 72 that is capable of navigating
5 hyperlinks and receiving and rendering hypermedia content, such as HTML pages.
6 As an alternative to a browser, the client may run a piece of software known as
7 "WebTV for Windows.", which is available from Microsoft Corporation.

8 In addition to the operating system 70 and browser 72, the client 24(1) is
9 illustrated with a number of software programs or modules that can be started or
10 called to receive and play enhancing content in conjunction with streaming
11 content. The modules include an announcement listener 74, one or more filters 76,
12 one or more controls 78, and one or more scripts 80. Listener 74 is a specialized
13 control that operates on information provided by announcements to receive and
14 process triggers. Each program is stored in program memory 56, loaded into
15 volatile memory 54 when launched, and executed on the processor 52. These
16 software components are described below in more detail with reference to Fig. 3.

17 The client 24(1) also has maintains a TV system services (TSS) database 82
18 that holds an electronic program guide (EPG) and a spool directory 84 that
19 functions as a cache for the TV based services and to hold the enhancing data files.

20 21 **Client Software Architecture**

22 Fig. 3 shows an exemplary configuration of the client software architecture
23 90. The streaming and enhancing content may arrive at the client in one of two
24 ways. First, a single source (e.g., primary content provider 22(1)) may transmit
25 both the streaming content and the enhancing content in a composite data stream.

1 In traditional television broadcast, the enhancing content is inserted into the VBI.
2 Alternatively, the streaming content may be served from one source (e.g., primary
3 content provider 22(1)) while the enhancing content is served from a second
4 source (e.g., secondary content provider 22(2)).

5 The enhancing content comprises the three elements: announcements,
6 triggers, and data files. The three elements are sent at different times to different
7 IP addresses and ports. The data files include both the enhancement files and the
8 dependency files.

9 Receiver 64 receives the composite stream or the dual streams. Receiver 64
10 represents one or more receiver types, such as a broadcast receiver, a network
11 connection, a cable receiver, and so forth. The receiver extracts the streaming
12 content (e.g., video and audio data) and sends it to one or two types of video
13 controls 78: a video control, which is used in Web pages that are not hosted in
14 WebTV for Windows, and an enhancement video control, which is used in Web
15 pages that are hosted in WebTV for Windows. The video controls reside in an
16 HTML page 92 that is opened by a browser 72 (or the WebTV for Windows
17 software) to support the interactive program.

18 As video is received by the video control 78, elements of the enhancement
19 stream (announcements, triggers, data files) are received at other components of
20 the client. The enhancement announcements are passed through one or more filters
21 76, which examine each announcement for a match against a list of programs in
22 which the user is interested, or against other types of predefined rules of
23 acceptance. The filter(s) 76 retain the announcements of interest, and discard the
24 rest.

1 Selected announcements are stored in the TSS database 82, which also
2 contains program guide information. The announcements are correlated with the
3 video content shows that can be enhanced. In this manner, the presence of
4 enhancement data in conjunction with a show listed in the database indicates that
5 the show is interactive.

6 Enhancement listener 74 handles the announcements stored in the TSS
7 database 82 when the user requests the corresponding interactive program. The
8 listener 74 uses information from the announcements to prepare to receive triggers.
9 The enhancement listener 74 is implemented as an ActiveX control that responds
10 to enhancement stream events. Like the video controls, the enhancement listener
11 is hosted in an enhancement page 92 displayed by browser 72.

12 Depending upon the trigger type, the listener 74 may perform several
13 different functions. A data-type trigger is used in facilitating reception of the data
14 files. The data-type trigger contains the IP address and port at which the data files
15 will be delivered. The enhancing data files are temporarily cached in spool
16 directory 84.

17 A navigation-type trigger is used for navigation about the container HTML
18 page 92. A script-type trigger contains information to invoke one or more scripts
19 80 for execution. The scripts use the enhancing data stored in the spook directory
20 84 to enhancing the streaming content being played by the video control 78.

21 22 Content Synchronization

23 Fig. 4 shows exemplary steps in a method for synchronizing streaming
24 content with enhancing content through use of announcements and triggers. At
25 step 100, the client receives streaming and enhancing content from one or more

1 content providers. The receiver passes the streaming content to video control 78 in
2 an HTML page 72 (step 102).

3 Meanwhile, the enhancement listener 74 executes in the background to
4 listen for announcements that may accompany streaming content. One or more
5 filters 76 register with, and are loaded by, the announcement listener 74 to filter
6 out unwanted announcements received in the enhancing content stream (step 104).
7 At step 106, announcements surviving the filtering process are stored in TSS
8 database 82. The filter 176 calls a loader (e.g., DLL object) to load the
9 announcement in the TSS database. The selected announcements are stored in
10 correlation with the titles or identities of the streaming content programs to
11 indicate that the programs are interactive.

12 Each time a user tunes a new channel, browser 72 checks the TSS database
13 82 to see if the new show is interactive (step 108). If not (i.e., the "no" branch
14 from step 108), the client simply plays the streaming content (step 110). On the
15 other hand, if the new show is interactive (i.e., the "yes" branch from step 108) and
16 assuming the user has enhancements enabled, the client browser opens a container
17 HTML page 92 and displays a special icon indicating that the show is interactive
18 (step 112). The container page 92 includes the controls and scripts to render the
19 streaming and enhancing content on the display.

20 When browser 72 displays an interactive show, the listener 74 listens on the
21 IP address and port specified in an earlier announcement stored in the TSS
22 database 82 (step 114). The announcement listener 74 monitors the IP stream at
23 the address and port for triggers. Triggers are sent at specific times during the
24 associated streaming content to cause an action to occur on a client in relation to
25

1 the streaming content. In this manner, the enhancing content induced by the
2 triggers is synchronized with the streaming content.

3 When a trigger is received, the listener 74 processes the trigger (step 116).
4 Part of the processing determines whether the trigger is a standard trigger, such as
5 the Broadcast Architecture defined data-type, navigation-type, and script-type
6 triggers, or a user-defined trigger.

7 For a data-type trigger, the client is prepared to receive enhancing data files
8 according to a particular protocol (step 118). One example of a data-type trigger,
9 known as the FTS data trigger, indicates a subdirectory in which to store
10 enhancing data files in the spool directory 84 and the IP address and port on which
11 files will be transferred. The FTS data trigger uses a "bpc:" URL syntax to
12 reference delivered files from HTML. The syntax allows for the use of
13 subdirectories. A sample FTS data trigger is given as follows:

14
15 1 <FTS> 233.43.17.17:17817&MyEnhancement
16

17 The enhancing data files are stored in the MyEnhancement subdirectory of
18 the spook directory 84 (step 120). The URL to a referenced HTML page in the
19 MyEnhancement subdirectory is given as follows:

20
21 bpc://MyEnhancement/funstuff.htm
22

23 For a navigation-type trigger, the browser uses the trigger information to
24 perform general navigation operations, such as moving to a top of page or
25 referencing a target resource (step 122). For a script-type triggers and user-defined

1 triggers, the listener sends an event to the container Web page 92, where they are
2 handled by scripts 80 embedded in that Web page (step 124).

3 If the triggers involve accessing enhancement data files, the listener 74 uses
4 the URL to retrieve the data files from the spool directory 84 in timely manner
5 (step 126). The browser then renders the enhancing content in synchronization
6 with the streaming content (step 128).

7 8 **Trigger-Carried Ticker Information**

9 Another aspect of this invention concerns a way to receive headlines and
10 other information as triggers and accumulate them into a ticker. Triggers holding
11 ticker information are accumulated over time and then displayed in a ticker. It also
12 provides a way to have text fade-in and fade-out (using Dynamic HTML or
13 "DHTML").

14 To demonstrate this technology, Fig. 5 shows an HTML page 150 for the
15 MSNBC Web site "www.MSNBC.com" that combines and coordinates both
16 streaming content and enhancing content. It includes a video pane 152 within
17 which the streaming video data is played. The MSNBC page 150 also has a static
18 link 154, such as a link to a home page, weather, or local news. Dynamic links
19 156 are edited and managed on a more routine basis, such as daily or hourly, and
20 hence change more often than the static link 154.

21 The MSNBC page also has an advertising space 158 to display DHTML
22 pages used to present advertisements. A DHTML ticker 160 is provided to scroll
23 section names, headlines, URLs, schedule information, and so forth.

24 The MSNBC enhancement page 150 operates as follows:
25

1. The enhancement page 150 is received and rendered by the browser 72.
2. The enhancement page 150 receives triggers containing headlines and other items for the ticker 160 and stores them as elements in an array. The enhancement page 150 also receives triggers for the dynamic links 156.
3. The page operates on a timer to cycle through all of the stored triggers, fading from one headline to the next.
4. Each headline has an associated Web link (URL); if the user clicks on a headline, a child browser window is created in which the referenced page is displayed.

The ticker data is stored in three arrays defined as follows:

```
// MSNBC Headlines
MSNBC_Category = new MakeArray(40);
MSNBC_URL = new MakeArray(40);
MSNBC_Headline = new MakeArray(40);
```

The array "MSNBC_Category" stores the headline category; "MSNBC_URL" stores the link associated with the headline, and "MSNBC_Headline" stores the actual headline text that is displayed in the ticker. Each array can store up to forty elements corresponding to ticker headlines. The array is initially filled with empty strings. As the ticker cycles through the arrays, it ignores any element that is an empty string. To fill out the arrays and thereby add headlines to the ticker data, the content provider transmits a trigger that calls the following Jscript routine:

651E0"5828281

```
1      function ChangeHeadline(tnum,tcats,turl,thl)
2      {
3      tnum--;
4      if(tnum >= 0 && tnum < 40)
5      {
6          MSNBC_Category[tnum] = tcats;
7          MSNBC_URL[tnum] = turl;
8          MSNBC_Headline[tnum] = thl;
9      }
10     }
```

11 The "ChangeHeadline" function inserts the headline "thl" at the array index
12 "tnum". The value specified in "tnum" allows the content producer to order the
13 headlines in a particular fashion. The ticker cycles through the arrays in numeric
14 order, skipping empty array elements.

15 Here are some examples of triggers that call "ChangeHeadline". Trigger
16 type 4 causes the trigger listener control 74 on the enhancement page to execute
17 the script specified in the trigger.

```
18 00:00:05.00 trigger 4 (
19 "top.main.ChangeHeadline(1,'Brian','http://msw','T
20 est 1 Headline');" ) only;
```

```
21 00:00:10.00 trigger 4 (
22 "top.main.ChangeHeadline(10,'Seattle','http://msw'
23 , 'Test 10 Headline Test 2 Headline Test 2 Headline
24 Test 2 Headline Test 2 Headline Test 2 Headline
25 Test 2 Headline Test 2 Headline Test 2
Headline');" ) only;
```

```
00:00:15.00 trigger 4 (
"top.main.ChangeHeadline(11,'Seattle','http://msw'
, 'Test 11 Headline Test 2 Headline Test 2 Headline
Test 2 Headline Test 2 Headline Test 2 Headline
```

Test 2 Headline Test 2 Headline Test 2
Headline');") only;

The following trigger will remove the headline from the array at index 11.

00:00:15.00 trigger 4 (
"top.main.ChangeHeadline(11, '', '', '');") only;

The following is an excerpt from an actual stream file as generated by the
MSNBC tools:

1 steps described. Rather, the specific features and steps are disclosed as exemplary
2 forms of implementing the claimed invention.
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25